AMENDMENTS TO THE SPECIFICATION

Please amend the paragraph beginning on line 13 of page 8 as follows:

It is a second object of the present invention to provide a playback apparatus that can perform special playback functions such as forward and backward search for music contents recorded on a semiconductor memory card without using a large-capacity memory.

~~The first object of the present invention can be achieved by a semiconductor memory card that stores at least one audio track, including: a protected area that can be accessed by a device connected to the semiconductor memory card only if the device has been found to be authentic, the protected area storing an encryption key sequence composed of a plurality of encryption keys arranged into a predetermined order; and an unprotected area that can be accessed by any device connected to the semiconductor memory card, the unprotected area storing at least one audio track and management information, the at least one audio track including a plurality of encrypted audio objects, and the management information showing which encryption key, out of the plurality of encryption keys, corresponds to each audio object stored in the unprotected area.~~

a1

The first object of the present invention can be achieved by a semiconductor memory card that stores at least one audio track, including: a protected area that can be accessed by a device connected to the semiconductor memory card only if the device has been found to be authentic, the protected area storing an encryption key sequence composed of a plurality of encryption keys arranged into a predetermined order; and an unprotected area that can be accessed by any device connected to the semiconductor memory card, the unprotected area storing at least one audio track and management information, the at least one audio track including a plurality of encrypted audio objects, and the management information showing which encryption key, out of the plurality of encryption keys, corresponds to each audio object stored in the unprotected area.

Please amend the paragraph beginning on line 14 of page 13 as follows:

These and other objects, advantages and features of the invention will become apparent from the following description thereof taken in conjunction with the accompanying drawings which illustrate a specific embodiment of the invention. In the Drawings:

FIG. 1 shows the appearance of a flash memory card 31 when viewed from above;

FIG. 2 shows the construction of the flash memory card 31 when viewed from below;

FIG. 3 shows the hierarchical composition of the flash memory card 31 in the embodiments;

FIG. 4A shows the special region, the authentication region and the user region provided in the physical layer of the flash memory card 31;

FIG. 4B shows the composition of the authentication region and the user region in the file system layer;

FIG. 5 shows the detailed composition of the file system layer;

FIG. 6 is a representation of when the AOB file "AOB001.SA1" is divided into five parts that are stored in clusters 003, 004, 005, 00A, and 00C;

a2

FIG. 7 shows one example of the settings of the directory entries and file allocation table when the AOB file "AOB001.SA1" is recorded in a plurality of clusters;

~~FIGs. 8A and 8B~~ FIGS. 8A and 8B show what directories are provided in the user region and the authentication region in the file system layer when the above two types of data are recorded in the application layer, as well as what kind of files are recorded in which directories;

FIG. 9 shows the correspondence between the file "AOBSA1.KEY" and the AOB files in the SD_Audio directories;

FIG. 10 shows the hierarchical composition of the data in an AOB file;

FIG. 11A shows the parameters stipulated by ISO/IEC 13818-7 standard in tabular form;

FIG. 11B shows the parameters that should be used when encoding a file in MPEG-Layer 3 (MP3) format in tabular form;

FIG. 11C shows the parameters that should be used when encoding a file in Windows Media Audio (WMA) format in tabular form;

FIG. 12 shows the detailed construction of an AOB_FRAME;

FIG. 13 shows how the byte length of the audio data in each of three AOB_FRAMEs is set;

FIG. 14 shows the correspondence between the sampling_frequency and the number of AOB_FRAMEs included in an AOB_ELEMENT;

FIG. 15 shows examples of the playback periods of AOB_ELEMENTs and the playback periods of AOB_FRAMEs;

FIG. 16 shows what is reproduced when the AOBs and AOB_BLOCKs recorded in an AOB file are consecutively played back;

FIG. 17 shows the hierarchical composition of the PlaylistManager and TrackManager used in the embodiments in detail;

FIG. 18 shows the sizes of the PlaylistManager and the TrackManager;

FIG. 19 shows the correspondence between the TKIs shown in FIG. 17 and the AOBs and AOB files shown in FIG. 16;

FIG. 20 shows the detailed data composition of the TKTMSRT shown in FIG. 17;

a2

FIG. 21 shows one example of the TKTMSRT;

FIG. 22 shows the detailed composition of the TKGI;

~~FIGs. 23A and 23B~~ FIGS. 23A and 23B show the composition of the BIT;

FIG. 23C shows the Time_Length field;

FIG. 24 shows cluster 007 to 00E into which the AOB composed of AOB_ELEMENT#1 to AOB_ELEMENT#4 are stored;

FIG. 25 shows how the next AOB_FRAME#x+1 to be played back is set when forward search is performed starting from the AOB_FRAME#x in an arbitrary AOB_ELEMENT#y in an AOB;

~~FIGs. 26A and 26B shows how~~ FIGS. 26A and 26B show how an AOB, an AOB_ELEMENT, and an AOB_FRAME that correspond to an arbitrary playback time code are specified;

FIGS. 27A and 27B show the deletion of a track;

FIG. 28A shows the TrackManager after the deletion of a track has been performed several times;

FIG. 28B shows how a new TKI and AOB file are written when "Unused" TKIs are present in the TrackManager;

FIGS. 29A and 29B show the TKIs are set when two tracks are combined to produce a new track;

FIG. 30A shows a Type1 AOB;

FIG. 30B shows Type2 AOBs;

FIG. 31A shows the combining of a plurality of tracks into a single track for a combination of a Type1+ Type2+ Type2+ Type1 AOB;

FIG. 31B shows the combining of a plurality of tracks into a single track for a combination of a Type1+ Type2+ Type2+ Type2+ Type1 AOB;

FIG. 32A shows a pattern where a Type1 AOB is present at the end of a preceding track and a Type1 AOB is present at the start of a next track;

a2

FIG. 32B shows a pattern where a Type1 AOB is present at the end of a first track and a Type2 AOB is present at the start of a next track;

FIG. 32C shows a pattern where a Type1 and Type2 AOB are present at the end of a first track and a Type1 AOB is present at the start of a next track;

FIG. 32D shows a pattern where a Type1 and Type2 AOB are present at the end of a first track and a Type2 and a Type1 AOB is present at the start of a next track;

FIG. 32E shows a pattern where two Type2 AOBs are present at the end of a first track and a Type1 is present at the start of a next track;

FIGS. 33A and 33B show the division of a track to produce two tracks;

FIGS. 34A and 34B show the content of the SD_Audio directory entries in the SD_Audio directory including the AOB file "AOB003.SA1" before and after the division of the track;

FIG. 35A shows the division of an AOB midway through AOB_ELEMENT#2;

FIG. 35B shows the two AOBs, AOB#1 and AOB#2, obtained by dividing an AOB midway through AOB_ELEMENT#2;

FIG. 36 shows how the BIT is set when an AOB is divided as shown in FIG. 35;

FIG. 37 shows a specific example of changes in the BIT before and after division;

FIG. 38 shows a specific example of changes in the TKTMSRT before and after division;

FIG. 39A shows the format of a DPL_TK_SRP;

FIG. 39B shows the format of a PL_TK_SRP;

FIG. 40 shows the interrelation between the Default_Playlist_Information, the TKIs, and the AOB files;

FIG. 41 shows example settings for the Default_Playlist and several PLIs;

FIG. 42 shows how the DPL_TK_SRPs correspond to TKIs using the same notation as FIG. 40;

FIGS. 43A and 43B show how the order of tracks is rearranged;

FIGS. 44A and 44B show how the Default_Playlist, TrackManager, and AOB files will be updated when DPL_TK_SRP#2 and TKI#2 are deleted from the Default_Playlist shown in FIG. 40;

a2

FIGS. 45A and 45B show how a new TKI and DPL_TK_SRP are written when an "Unused" TKI and DPL_TK_SRP are present;

FIGS. 46A and 46B show how tracks are combined;

FIGS. 47A and 47B ~~shows how~~ show how a track is divided;

FIG. 48 shows the appearance of a portable playback apparatus for the flash memory card 31 of the present embodiments;

FIG. 49 shows one example of the display on the LCD panel when a playlist is selected;

FIGS. 50A to 50E show examples of the display on the LCD panel when a track is selected;

FIGS. 51A to 51C show example operations of the jog dial;

FIG. 52 shows the internal construction of the reproduction apparatus;

FIG. 53 shows how data is transferred in and out of the double buffer 15;

~~FIG. 54A and 54B shows how~~ FIGS. 54A and 54B show how areas in the double buffer 15 are cyclically allocated using ring pointers;

FIG. 55 is a flowchart showing the AOB file read procedure;

FIG. 56 is a flowchart showing the AOB file output procedure;

FIG. 57 is a flowchart showing the AOB file output procedure;

FIG. 58 is a flowchart showing the AOB file output procedure;

FIGS. 59A to 59D show how the playback time code displayed in the playback time code frame on the LCD panel 5 is updated in accordance with the updating of the variable Play_time;

FIG. 60 is a flowchart shows the processing of the CPU 10 when the forward search function is used;

FIGS. 61A to 61D show how the playback time code is incremented when the forward search function is used;

FIGS. 62A and 62B show specific examples of how the time search function is used;

FIG. 63 is a flowchart showing the processing in the editing control program;

FIG. 64 is a flowchart showing the processing in the editing control program;

FIG. 65 is a flowchart showing the processing in the editing control program;

a2

FIG. 66 shows one example of a recording apparatus for recording data onto the flash memory card 31;

FIG. 67 shows the hardware configuration of the recording apparatus;

FIG. 68 is a flowchart showing the processing during recording;

FIG. 69 shows the hardware construction of the flash memory card 31;

FIG. 70 shows the communication sequence used when a playback apparatus connected to the flash memory card 31 reads the encryption key FileKey and plays back AOBs; and

FIG. 71 shows the details of the communication sequence used when mutual authentication is performed in FIG. 70.

Please amend the paragraph beginning on line 24 of page 25 as follows:

a3

FIG. 4B shows the construction of the authentication region and the user region in the file system layer. As shown in FIG. 4B, the authentication region and the user region in the file system each include "partition boot sectors", a "file allocation table (FAT)", a "root directory", and a "data region", meaning that the authentication region and the user region have the same construction. FIG. 5 shows the various parts of these file systems in more detail. The following describes the construction of the user region with reference to FIGS. 4A, 4B and 5.

a3 ~~shows the various parts of these file systems in more detail. The following describes the construction of the user region with reference to FIGS. 4A, 4B and 5.~~

Please amend the paragraph beginning on line 24 of page 47 as follows:

a4 As shown in FIG. 17, the TrackManager is composed of the Track Information (TKI) #1, #2, #3, #4 . . . #n, as shown by the broken line h1. These TKIs are information for managing the AOBs recorded in AOB files as tracks, and each correspond to a different AOB file. From FIG. 17, it can be seen that each TKI is composed of Track_General_Information (TKGI), Track_Text_Information ~~(TKTXTI_DA)~~ (TKTXI_DA) in which text information exclusive to a track can be written, and a Track_Time_Search_Table (TKTMSRT) that serves as a time search table.

Please amend the paragraph beginning on line 9 of page 48 as follows:

a5 From FIG. 18, it can be seen that each TKI has a fixed size of 1,024 bytes, which means that total size of the TKGI and the ~~TKTXTI_DA~~ TKTXI_DA is fixed at 512 bytes due to the size of the TKTMSRT being fixed at 512 bytes. In the TrackManager, a total of 999 TKIs can be set.

Please amend the paragraph beginning on line 15 of page 82 as follows:

a6 The following describes how the information fields in the TKI are set for the AOB file "AOB002.SA1" once this file has been obtained by updating the directory entries and the FAT values. When generating a TKI for a divided track, there are two kinds of information fields in the TKI. These are (1) information that can be copied from the original TKI and (2) information obtained by updating the information in the original TKI. The ~~TKTXTI_DA~~ TKTXI_DA and ISRC are the former type, while the BIT, the TKTMSRT and other information fields are the latter type. Since both types of information exist, the present embodiment generates a TKI for a divided track by copying the original TKI to produce a template for the new TKI, and then dividing/updating the TKTMSRT and BIT in this template and updating the remaining information fields.

Please amend the paragraph beginning on line 10 of page 85 as follows:

a7 As can be seen from FIG. 36, the division of the AOB result in only the AOB_ELEMENT that includes the boundary for the division being divided into two and in the other AOB_ELEMENTs positioned before and after the divided AOB_ELEMENT remaining unchanged. As a result, the "FN_Last_TMSRTE" of AOB#2 is set at the same value for the ~~"AOB_ELEMENT#4"~~ "AOB_ELEMENT#3" before the division, and the "FNs_1st_TMSRTE" of AOB#2 is set at AOB_ELEMENT#1 of AOB#2, which is to say, the number of frames included in the part that follows the boundary once AOB_ELEMENT#2 has been divided.

Please amend the paragraph beginning on line 10 of page 109 as follows:

a8 When the user rotates the jog dial counterclockwise as shown in ~~FIG. 51B,~~ FIG. 51C, the playback time code is reduced to "0:00:10" in keeping with the amount by which the jog dial was rotated. Conversely, when the user rotates the jog dial clockwise as shown in ~~FIG. 51C,~~ FIG. 51B, the playback time code is increased to "0:00:30" in keeping with the amount by which the jog dial was rotated.

Please amend the paragraph beginning on line 16 of page 119 as follows:

a9 Once the first AOB_FRAME has been outputted to the descrambler 7, in step S26 the playback apparatus refers to the ADTS header of AOB_FRAME#x and specifies where the next AOB_FRAME is located. In step S27, the playback apparatus increments the variable#x (#x←#x+1) and sets AOB_FRAME#x as the next AOB_FRAME. In step S28, AOB_FRAME#x is inputted into the descrambler 7. After this, in step S29, the variable play_time is incremented by the playback period of the AOB_FRAME#x and the variable play_data is incremented the amount of data corresponding the AOB_FRAME#x. After incrementing AOB_FRAME#x, in step S30 the CPU 10 judges whether the variable #x has reached the value given in FNs_1st_TMSRTE.